NASA Technical Memorandum 101730

Real-Time Application of Advanced Three-Dimensional Graphic Techniques for Research Aircraft Simulation

Steven B. Davis

December 1990

National Aeronautics and
Space Administration

NASA Technical Memorandum 101730

Real-Time Application of Advanced Three-Dimensional Graphic Techniques for Research Aircraft Simulation

Steven B. Davis
Ames Research Center
Dryden Flight Research Facility
Edwards, CA 93520-0273

1990

National Aeronautics and
Space Administration
Ames Research Center
Dryden Flight Research Facility
Edwards, California 93523-0273

CONTENTS

ABSTRACT

Visual aids are valuable assets to engineers for design, demonstration, and evaluation. This report discusses a variety of advanced three-dimensional graphic techniques used to enhance the displays of test aircraft dynamics. The new software's capabilities are examined and possible future uses considered.

NOMENCLATURE

C	color
DAFI	dynamic aircraft flight imaging
IEEE	Institute of Electrical and Electronic Engineers
IP	internet protocol
IRIS	interactive raster imaging system
N_b	the bisect vector
N_p	normal vector to the surface at point P
N_{pl}	vector from point P to the light P_l
Nx	normal vector parallel to x-axis
Ny	normal vector parallel to y-axis
Nz	normal vector parallel to z-axis
OFF	object file formatting
P	point
P_l	light source
RGB	red-green-blue
RPV	remotely piloted vehicle
TCP	transmission control protocol
UDP	user datagram protocol

INTRODUCTION

Viewing strip charts has been a common method of monitoring simulation and flight data. Strip chart readouts do not provide readily clear pictures of the complex dynamic motions of aircraft. Computer imaging can improve the analysis of test data by using visual aids. Graphic display software is useful because it can simulate the real world. Physical laws and natural phenomena are integrated into the software coding as algorithms of mathematical equations and calculations. Only after realistic simulations are provided can graphic display software be useful to engineers and scientists. The simulations are provided by applying computer graphic techniques.

A new approach to applying computer graphic techniques was used to create a program to enhance the display of the dynamic characteristics of research aircraft. The program runs on the interactive raster imaging system (IRIS) which uses various graphic techniques for the drawing, shading, and lighting of a displayed X-29 aircraft model. The software is driven by real-time data from the simulation computer. Dynamic velocity vector components (fig. 1) are juxtaposed on the X-29 model. The model is dynamically driven by three-axes attitude data, so the engineer immediately knows what the aircraft is doing. An engineer can determine the angle of attack, the angle of sideslip, and the velocity of the aircraft, among other uses. This paper describes the system and program used to show the motions of the X-29 aircraft.

SYSTEM OVERVIEW

The development of graphic display software entailed two considerations, how the software would function and which hardware to use. The software used to display and manipulate a three-dimensional, filled-polygon model of an X-29 forward-swept-wing aircraft was comprised of "C" language routines referred to as dynamic aircraft flight imaging (DAFI). The routines of the DAFI package were obtained from existing software packages used on the IRIS workstation. The database for the three-dimensional, X-29 display came from a demonstration program provided on the IRIS. The routines, derived from MYLIGHT (written by Michael J. Zyda of the Naval Postgraduate School, Monterey, CA), read the model data from various files and supported translation and rotation of the model. Data for the X-29 simulator is processed in an Encore/Gould/S.E.L. Concept 32/9780 computer. The simulation computer processes specified aerodynamic parameters by converting Gould formatted floating point numbers to Institute of Electrical and Electronic Engineers (IEEE) floating point standards. The processed data is directed as an 80-Hz output to the IRIS workstation, which can display graphics at a 60-Hz screen-clear rate. The parameters are used within the algorithms for the drawing, shading, and lighting of the displayed image.

THREE-DIMENSIONAL GRAPHIC TECHNIQUES

The rapid advancement of computer technology and high-quality graphics systems has provided today's engineer with the ability to display images which before could only be imagined. The IRIS provides users with the ability to produce basic two-dimensional and highly realistic three-dimensional images. The two-dimensional displays rely on a screen x- and y-coordinate system, with the positive x-axis directed from the left to the right side of the screen and the positive y-axis directed from the bottom to the top of the screen. Three-dimensional displays use a z-axis which has a positive direction out from the screen.

One technique used by the IRIS for three-dimensional imaging is hidden-surface removal. Hidden-surface removal involves displaying only the surfaces of a three-dimensional object which could be seen and not displaying surfaces out of eyesight. Two algorithms for hidden-surface removal are z-buffering and backfacing-polygon removal. The z-buffering operation calculates the distance from the surfaces covering each pixel to the eye and draws only the surface that is closest. The raster subsystem assigns a screen z-coordinate value to each pixel used to fill a polygon. Before a pixel is written into the frame buffer, the previous z-coordinate used to set that location is checked. If the new point is closer than any previous point, then the color in the frame buffer is changed and the z-coordinate in the z-buffer is changed to reflect the new, closer point. Any new points further away are discarded (*GT Graphics Library User's Guide*, Silicon Graphics, Mountain View, CA).

Another hidden-surface removal technique is backfacing polygon removal. In this technique, each polygon face of an object is drawn in counterclockwise order, as viewed from outside the object. If the object is rotated halfway around the y-axis, the faces on the front will be in counterclockwise order and those on the back will be in clockwise order. The IRIS system can be put in a mode where only counterclockwise polygons are drawn, so the hidden surfaces of a three-dimensional object will be removed (as described in the *GT Graphics Library User's Guide*). Because the z-buffering algorithm is more reliable, it was used instead of backfacing polygon removal. Figure 2 shows the basic operation of each hidden-surface-removal algorithm.

The shading characteristics of an image improve its realism and appearance. For two-dimensional, uniformly colored objects, flat shading may be applied. For curved surfaces, colors vary continuously across the surface, so a flat-shaded polygonal approximation to the surface looks tiled. Gourad shading, also known as "intensity interpolation shading," makes light intensity smooth across polygons. Gourad shading accommodates the depth expectancy of three-dimensional objects, so it was used for program development rather than flat facet shading. Figure 3 shows the graphical quality improvement of gourad shading over flat shading. The Gourad shading is accomplished by:

1. Calculating the surface normals for each polygon, which determines all normal vectors for each polygon of the model's surface,

2. calculating the normal vectors for shared vertices between adjacent polygons by simple vector addition,

3. calculating the shading intensities for each vector value, allowing the boundaries between adjacent polygons to be concealed by the shading, and

4. assigning shading over the rest of the polygons by linear interpolation according to the normal vector values which constituted each vertex vector.

The lighting model facility on the IRIS automatically calculates color using the geometries, colors, and properties of the current material, lights, and lighting model. One material, eight lights, and one lighting model are selected and bound as inputs to a lighting equation. The lighting equation determines the color at a point based on the normal and material properties and point's position, the position and properties of the light sources, and the lighting model properties. The lighting equation is evaluated when the normal or graphics position of an object changes, depending on whether the light sources and view are local. The basic equation used for lighting in DAFI is:

$$C_p = C_{emitted\,light} + C_{ambient\,reflected\,light} + C_{diffuse\,reflected\,light} + C_{specular\,reflected\,light} \tag{1}$$

where C_p is the color of the point. The other components of the equation are:

$C_{emitted\,light}$ is the emission color of the material. Associated with self-luminous materials.

$$C_{ambient\,reflected\,light} = C_{sa} * C_{ma} + C_{la} * C_{ma} \tag{2}$$

where C_{sa} is the color of ambient light in the scene, C_{ma} is the ambient color of the material, C_{la} is the ambient color of the light, and ambient light is the scattered light in a scene.

$$C_{diffuse\,reflected\,light} = C_l * C_{md} * (N_{pl} \bullet Np) \tag{3}$$

where C_l is the color of the light, C_{md} is the diffuse color of the material, N_{pl} is the direction from point P to the light P_l, N_p is the normal to the surface at point P, and diffuse light is directed light which is reflected uniformly in all directions. Figure 4 shows the symbology representation.

$$C_{specular\,reflected\,light} = C_l * C_{ms} * [N_p \bullet N_b]^{Emss} \tag{4}$$

where

$$N_{\underline{b}} = (N_{\underline{pe}} + N_{\underline{pl}}) \,/\, |N_{\underline{pe}} + N_{\underline{pl}}|, \tag{5}$$

and C_{ms} is the specular color of the material, N_b is the bisector vector, N_{pe} is the direction from point P to the eye P_e, $Emss$ is the material's specular scattering component, and specular light is directed light that is reflected nonuniformily depending on direction. Figure 5 shows the symbology representation.

Figure 6 shows the process the IRIS used for a lighting calculation. Users can control the color, location, and direction of the light source or sources; the color and surface properties of the object; and the position and view direction of the observer.

X-29 DISPLAY ROUTINES

Many computer generated images cannot show the dynamic capabilities of components of the object. For instance, an X-29 model should demonstrate mobility of the flaps, strakes, canards, and rudder, based on simulation surface position. The IRIS system can dynamically manipulate individual surfaces of displayed models. The ability to view attitudes and dynamic surface positioning helps engineers monitor and verify aircraft capabilities.

The components of the X-29 model are stored in several data files (one file contains data to draw the canards, another the rudder, etc.). The use of "component" data files allows the X-29 model to have dynamic surfaces. For example, if information is sent from the simulator to the IRIS with data simulating a positive 30° rotation of the left canard, then the displayed model will respond by rotating its left canard by positive 30°. Using component data files also allows for future program expansion; if the landing gear position needs to be shown, for example, then data representing the gear could be placed into data files and graphically displayed by DAFI.

Object file formatting (OFF) is one particularly useful capability of the IRIS. OFF groups together sequences of drawing routines and uses identifiers as headers. When a reference is made to an identifier, a series of drawing routines is initiated. As used by the X-29 display routines, OFF follows these steps:

1. a component data file is created containing polygon surface definitions, such as coloration, data points, and origin location,
2. an OFF reader file reads the definitions with the aid of include files containing global system definitions and variables, operational codes, and other indices, and
3. the main program assigns identifiers to each component and references them for display when appropriate.

Velocity vector components which allow the user to determine airspeed, angle of attack, and flight direction are included in the display (fig. 7). Each component originates from the nose of the aircraft model and is color coded for easy identification and determination of the aircraft's velocity in any of the x-, y-, or z-axes of earth-referenced three-dimensional space. Using the parameters of the aircraft's velocity, angle of sideslip, and angle of attack, the overall velocity is broken into its various components and then relinked to form a "velocity viewing window." This window structure is better suited for three-dimensional viewing than a two-dimensional vector line, allowing easier data assimilation and display.

MODEL MANIPULATION ROUTINES

Matrices and matrix multiplication are used to accomplish the three-dimensional transformations used in DAFI. DAFI uses a right-handed, homogeneous coordinate system to support its transformation operations.

Objects are translated to new positions by adding translation amounts to the coordinates of each point of the objects. Using a homogeneous coordinate system allows simple matrix multiplication, rather than matrix addition, to be used to do the translation, represented by:

$$[x'y'z'1] = [xyz1] \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ dx & dy & dz & 1 \end{vmatrix} \tag{6}$$

yielding the linear equations:

$$x' = x + dx, \tag{7}$$

$$y' = y + dy, \tag{8}$$

$$z' = z + dz. \tag{9}$$

Objects are rotated to new positions by rotating their component points through an angle Θ about an assigned origin. Though rotation by matrix multiplication does not require homogeneous coordinates, the homogeneous system is applied for uniformity of matrix operations. Rotation about a given axis is calculated as:

$$[x'y'z'1] = [xyz1] \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\Theta & \sin\Theta & 0 \\ 0 & -\sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{10}$$

yielding:

$$x' = x \tag{11}$$

$$y' = y\cos\Theta - z\sin\Theta \tag{12}$$

$$z' = y\sin\Theta + z\cos\Theta \tag{13}$$

about the x-axis,

$$[x'y'z'1] = [xyz1] \begin{vmatrix} \cos\Theta & 0 & -\sin\Theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\Theta & 0 & \cos\Theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{14}$$

yielding:

$$x' = x\cos\Theta + z\sin\Theta \tag{15}$$

$$y' = y \tag{16}$$

$$z' = -x\sin\Theta + z\cos\Theta \tag{17}$$

about the y-axis, and

$$[x'y'z'1] = [xyz1] \begin{vmatrix} \cos\Theta & \sin\Theta & 0 & 0 \\ -\sin\Theta & \cos\Theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{18}$$

yielding:

$$x' = x\cos\Theta - y\sin\Theta \tag{19}$$

$$y' = x\sin\Theta + y\cos\Theta \tag{20}$$

$$z' = z \tag{21}$$

about the z-axis (Gemini Technology Corporation document no. GE881022, *Generic Visual System (GVs) Ethernet Local Area Network Communications*).

Positive rotation about each axis is:

Axis of Rotation	Direction of Positive Rotation
x	y to z
y	z to x
z	x to y

COMMUNICATIONS ROUTINES

An Ethernet link between the simulator and the workstation is used for data transference (described by Marlin Pickett of NASA Ames-Dryden in an unpublished document). Ethernet communication is established at the socket level. A socket is defined by a hostname (the machine to be contacted), a pair of addresses for socket location, and a protocol type, either transmission control protocol (TCP), user datagram protocol (UDP), or internet protocol (IP). The TCP is a connection-oriented, end-to-end reliable protocol designed to fit into a layered hierarchy of protocols supporting multinetwork applications. The TCP provides reliable interprocess communication between pairs of processes in host computers attached to distinct but interconnected computer communication networks. The UDP provides a procedure for application programs to send messages to other programs with a minimal mechanism. The protocol is transaction oriented, and delivery and duplicate protection are not guaranteed. The IP implements two functions: addressing and fragmenting. The internet modules use the addresses carried in the internet header to transmit datagrams toward their destinations. The internet modules use fields in the internet header to fragment and reassemble datagrams when necessary for transmission through "small packet" networks (*Generic Visual System (GVs) Ethernet Local Area Network Communications*). The UDP is fastest because its functioning is simple and it neglects error checking. Its speed was ideal for real-time application, so UDP, rather than IP or TCP, was used to develop DAFI. The socket is established by a background process associated with DAFI, which maintains shared memory. Shared memory allocates space that multiple processes can use. Once a socket is bound, the background process reads the data sent from the simulation computer and places the data into the shared-memory space. DAFI accesses the memory space and reads the latest available data. Using shared memory reduces latency associated with reading data directly from a buffer. When reading data from a buffer, an information overflow may occur. If data is placed in the buffer faster than the data can be read, a stack of old data equal to the buffer size can develop. While DAFI reads through the stack all new data are placed at the bottom and not used until reached. The result is a delay, or latency, in DAFI's response to external input and a hinder to real-time operation.

The background process also closes the communication link and removes the shared memory space when the DAFI program is terminated. The processing path followed is shown in figure 8. The parameters being sent (listed below) are all floating points.

Parameter	Units
Left canard position	deg
Right canard position	deg
Left flap position	deg
Right flap position	deg
Left strake position	deg
Right strake position	deg
Rudder postion	deg
Angle of attack	rad
Angle of sideslip	rad
Pitch attitude	rad
Roll attitude	rad
Heading angle	rad
Pitch stick position	in.
Roll stick position	in.

Parameter	Units
Rudder pedal	in.
Altitude	ft
Velocity	ft/sec
Acceleration in X direction	g
Acceleration in Y direction	g
Acceleration in Z direction	g
Time	sec after midnight

These parameters are used by the model manipulation routines to drive the display. Other parameters can be added as needed and integrated into the model manipulation routines.

Current Applications

For simulation purposes, DAFI can be used to display aircraft dynamics. Simulation manuevers which could be applied during an actual flight can therefore be designed. If pilots and ground-controllers know what dynamics to expect, they can conduct flights with reduced odds of undesirable occurrences.

Postflight analysis can be accomplished by using stored flight data as input parameters. Users can playback flight data and observe aircraft response in close detail. Playbacks can be done in slow-motion and from various viewing angles, allowing for a complete analysis of the aircraft. Investigations of aircraft dynamics and data comparisons can be evaluated and stored on video tape for future playbacks and reference.

Display position correlation matches the location of a displayed object on the screen with a specific parameter value. For example, the X-29 display would originally be centered on the screen. When the aircraft accelerates, the model would move from the center in the direction of the acceleration. Figure 9 shows a simple, two-dimensional example of the concept.

Future Applications

The DAFI can display any aircraft, because of the component data files it uses. A library of data files for various aircraft components could be built to facilitate this feature. A wide display capability would prevent DAFI from becoming obsolete.

Display position correlations need not be limited to accelerations. Correlations could also be made with regards to kinetic energy accumulation by the aircraft, induced lift, or other relations desired by the user.

In addition to providing a display for simulated aircraft performance, DAFI could serve as a real-time display in mission control for actual test flights. The display capabilities of DAFI may be valuable tools to test engineers for flight monitoring. The instrumented test aircraft would telemeter data to ground station computers, which would bus the received parameters through Ethernet to the IRIS workstation. Since the DAFI display could be used as a direct representation of an aircraft in flight, external malfunctions with the aircraft could be reflected in the display. Uncorresponding surface positioning and associated aircraft attitude could alert controllers of failure. If the aircraft control surfaces were not positioned as expected for a given pilot input or a particular flight condition then a failure occurrence would be noted. A dynamic aircraft display corresponding to actual attitude and surface positioning data can verify proper aircraft functioning faster than sorting through a strip chart or a list of parameters.

The IRIS color capabilities and resolution allow DAFI to do thermal modeling. A simplistic approach to thermal modeling might be:

1. Fragment aircraft component surface data files into smaller, discrete units.
2. Develop a file containing a color chart and associated color ranges.
3. Expand the input variable listing to accept data from temperature probes on the aircraft.
4. Provide a routine within the model display routines to monitor and assign appropriate coloration to each aircraft component, using the color file.

The accuracy of the coloration mapping would be limited only by the size of each component data. The smaller the data units, the more precise the display. Thermal modeling would be beneficial for hypersonic aircraft research, especially for individuals in the aerothermodynamics discipline. Thermal analysis and aircraft flight envelope study could be accomplished quickly by viewing a color thermal display.

Structural loads could be analyzed in the same way as thermal modeling. Strain gages on the aircraft would provide the parametric input and a color scheme assignment would differentiate the various input values. As loads are applied to the aircraft, a color mapping of the displayed model would allow users to determine quickly and accurately the structural integrity of various parts of the aircraft. This quick analysis would allow viewers to determine if a component is approaching failure.

Remotely piloted vehicles (RPVs) cannot always be watched closely during test flights. Though means are available for tracking RPVs, a visual display could be a useful addition. Erratic flight characteristics could indicate technical difficulties more quickly than sorting through numeric displays of relayed flight parameters.

CONCLUDING REMARKS

Computers have provided engineers and scientists a practical means of observing phenomena of interest. Dynamic aircraft flight imaging (DAFI) is the latest method of observation. The displayed images can relay data in a form more easily assimilated than strip charts and numeric displays. The use of DAFI is not limited to the X-29 aircraft. This software package can be a general display tool for many scientific and technological interests because of its intrinsic expansiveness and potential for varied applications.

Ames Research Center
Dryden Flight Research Facility
National Aeronautics and Space Administration
Edwards, California, August 14, 1990.

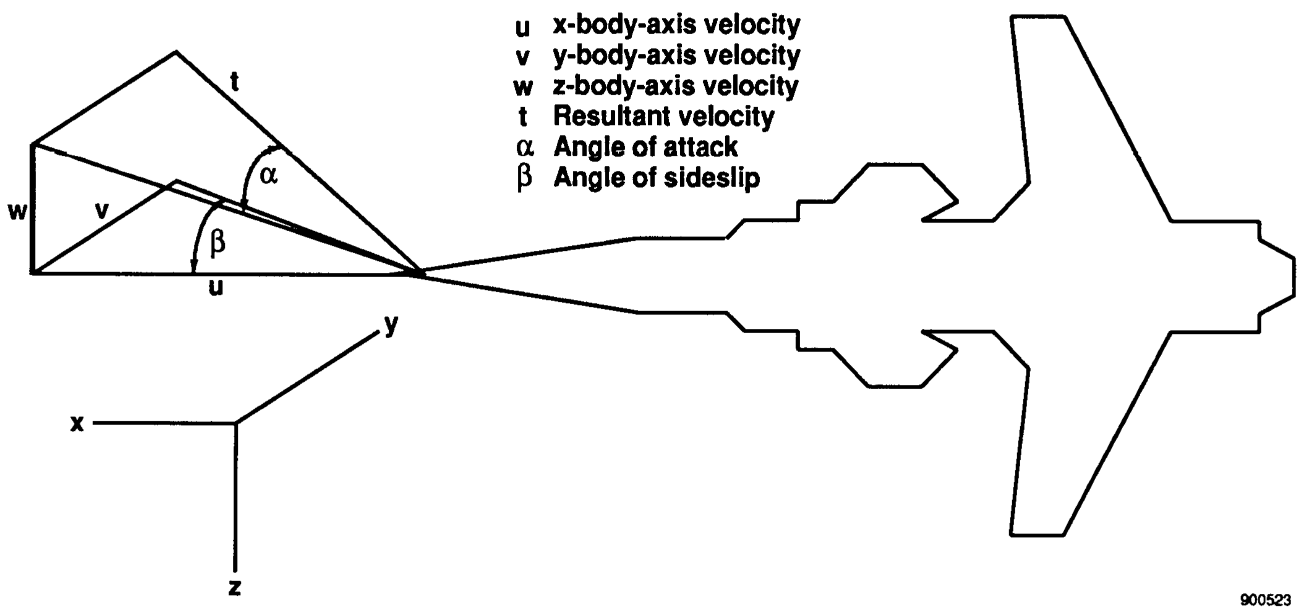
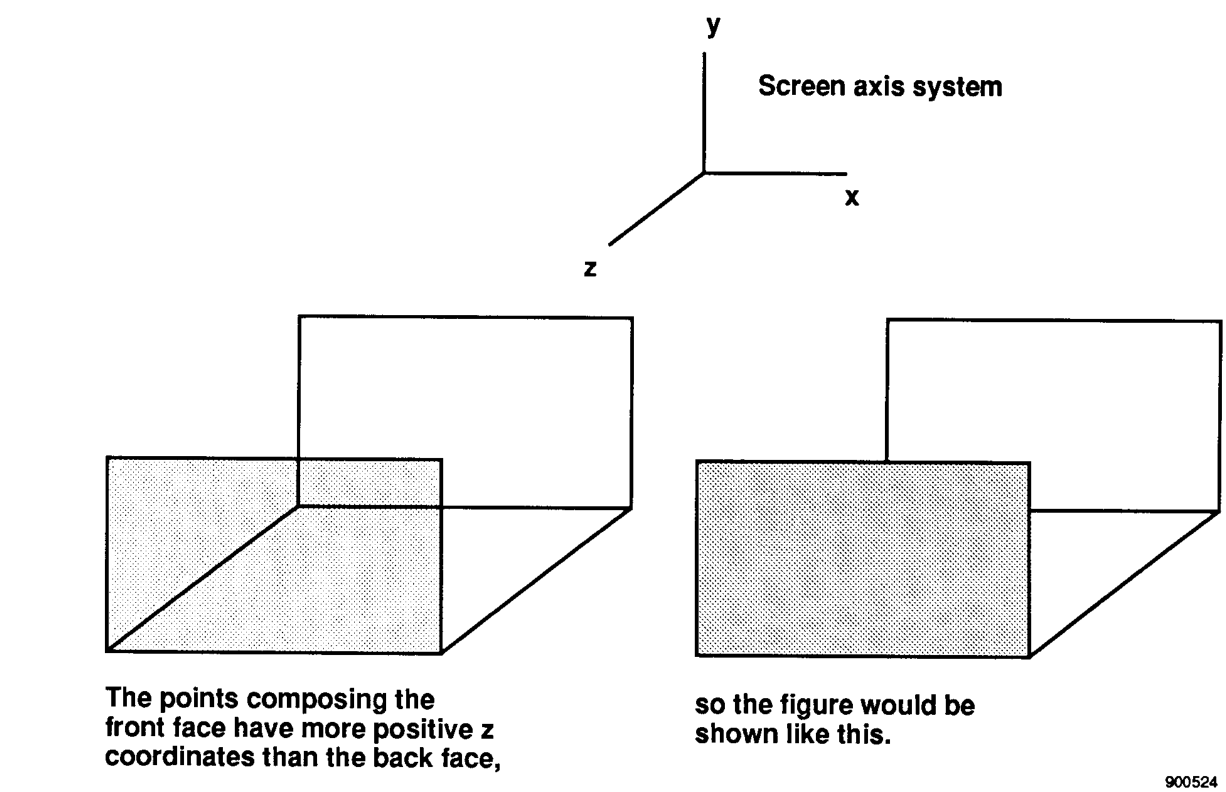
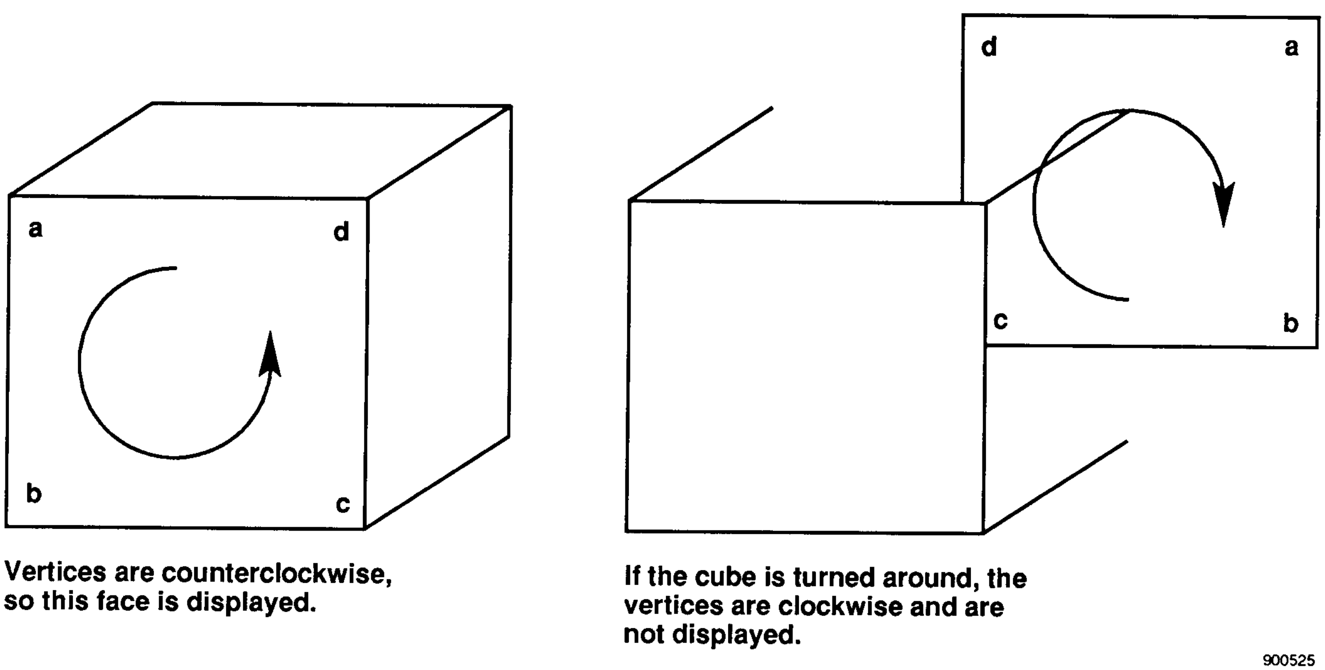


Figure 1. Velocity vector components.

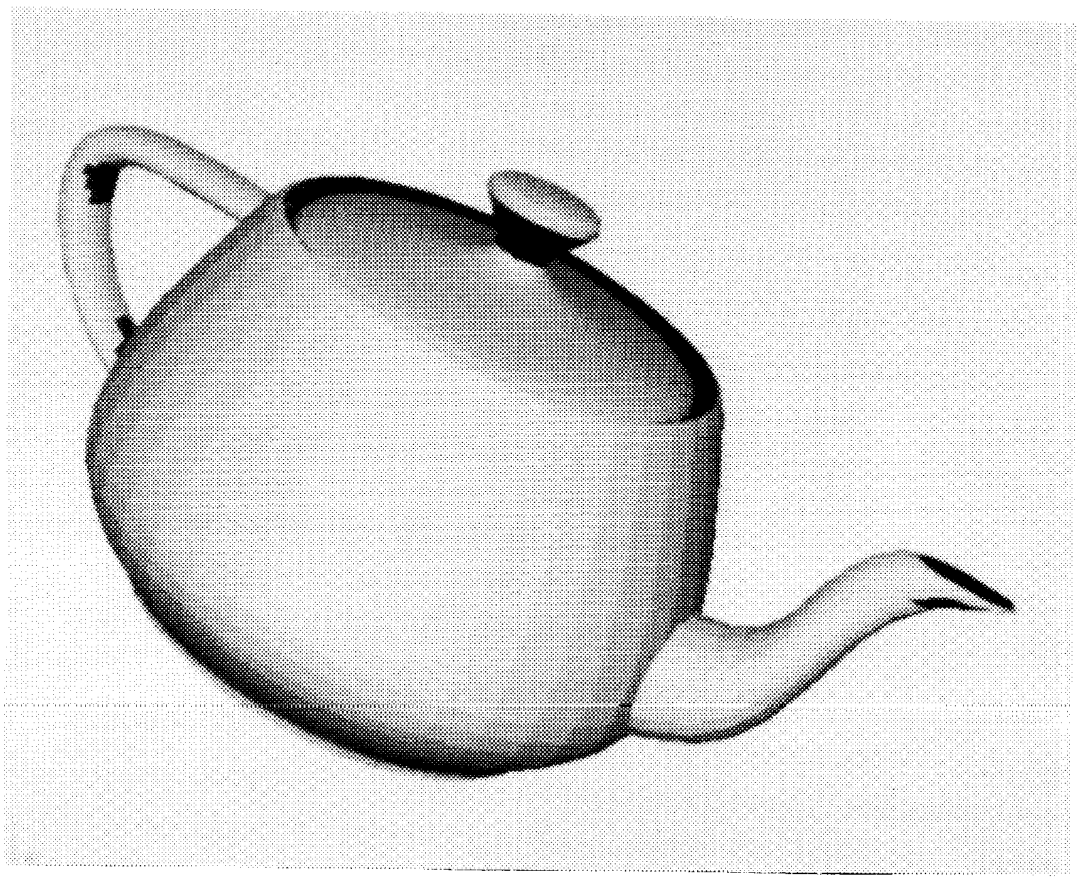


(a) The z-buffering technique.



(b) Backfacing polygon removal technique.

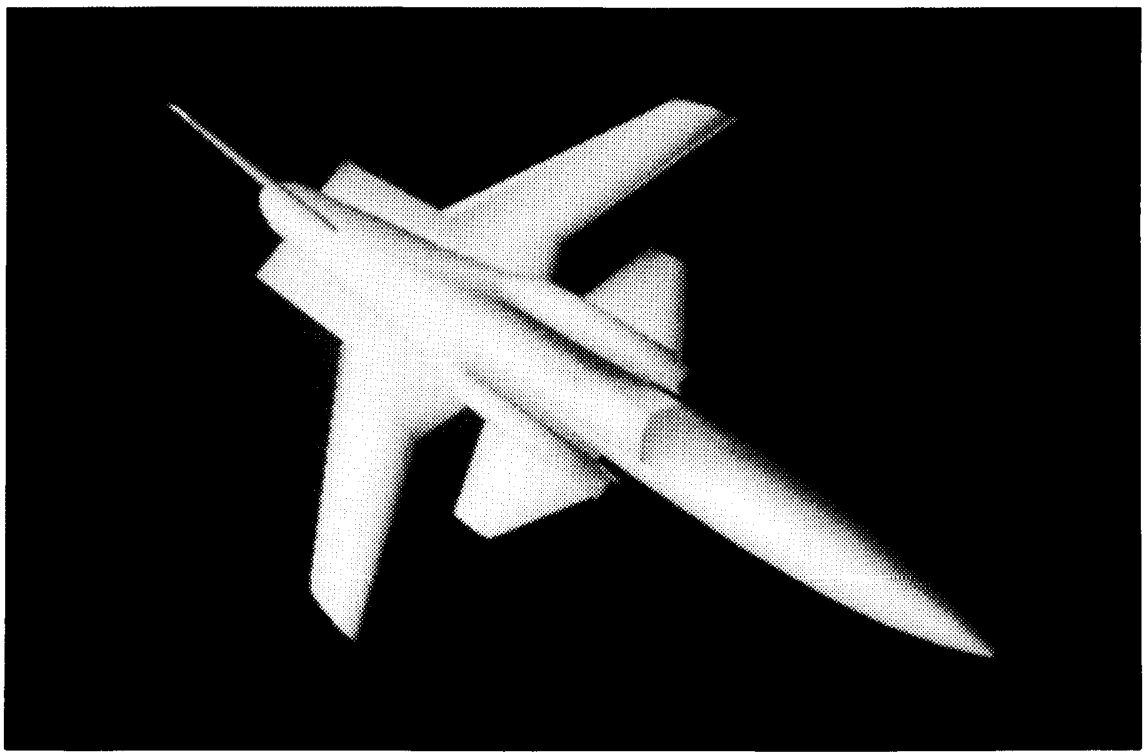
Figure 2. Hidden-surface removal.



EC90 0084-006

(a) Flat facet shading.

Figure 3. Shading techniques.



EC90 0084-004

(b) Gourad shading.

Figure 3. Continued

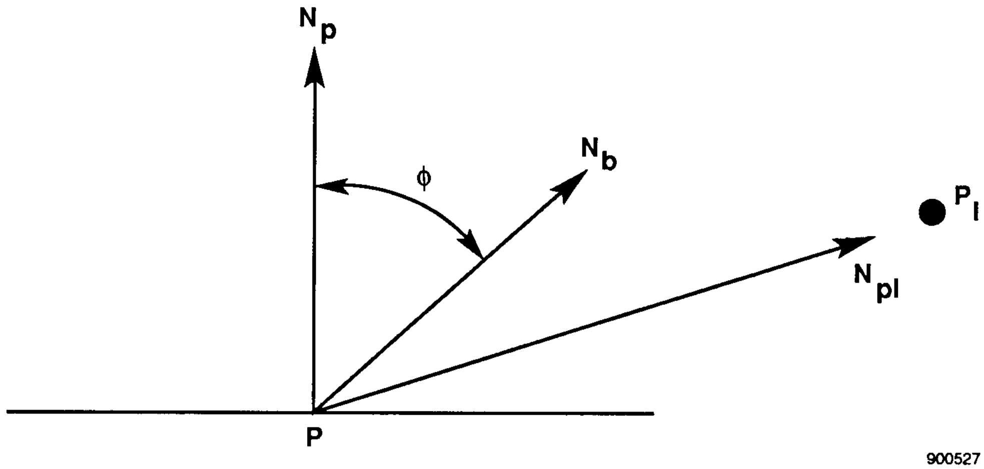


Figure 4. Diffuse light calculations basis.

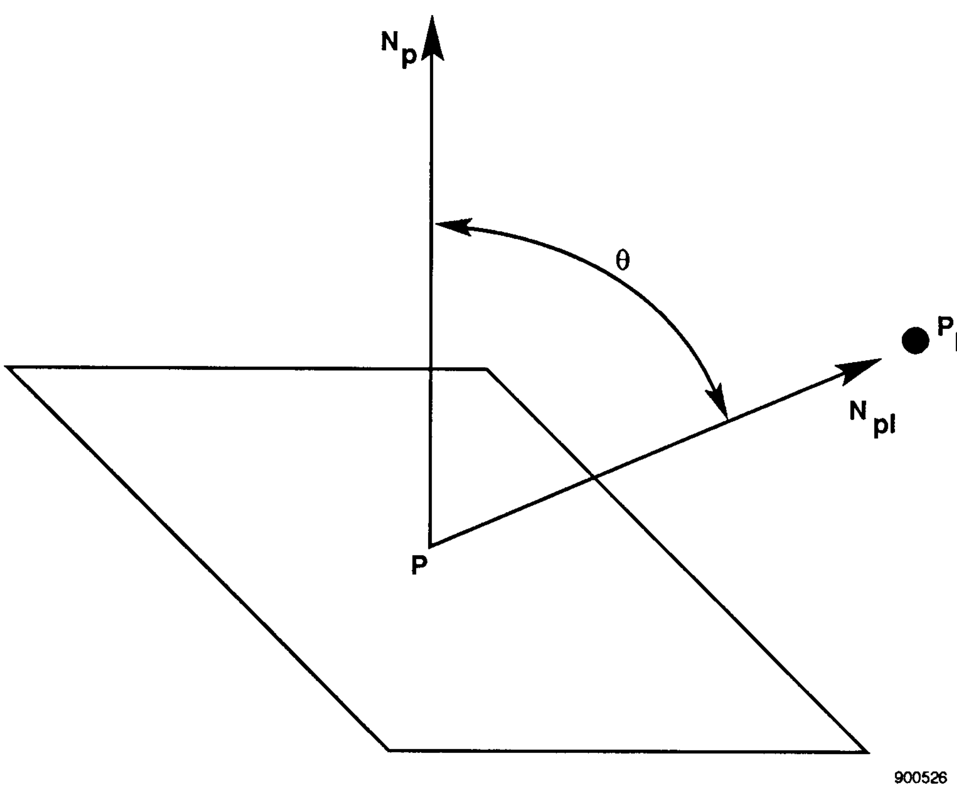


Figure 5. Specular light calculation basis.

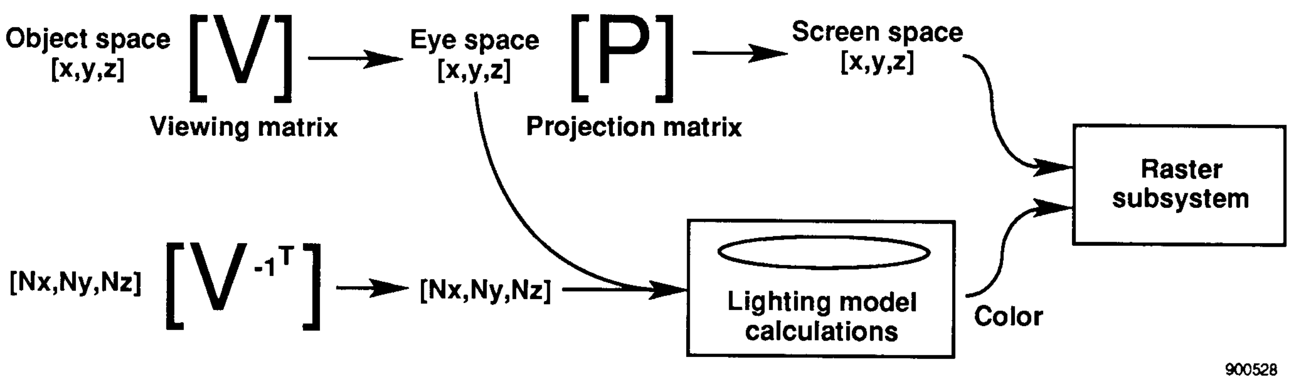
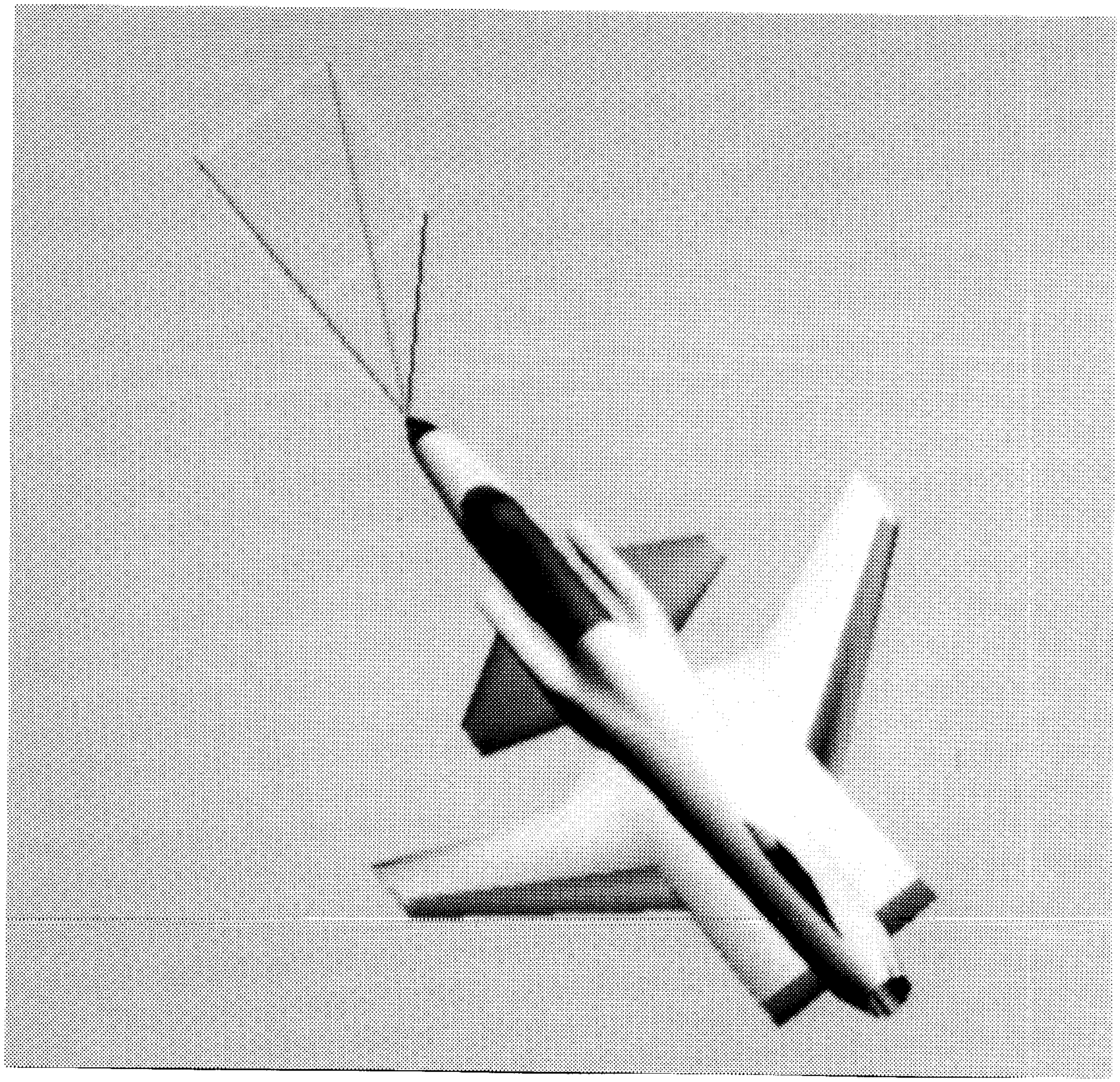


Figure 6. Lighting calculation.



EC90 0084-002

Figure 7. The DAFI model with velocity vector components.

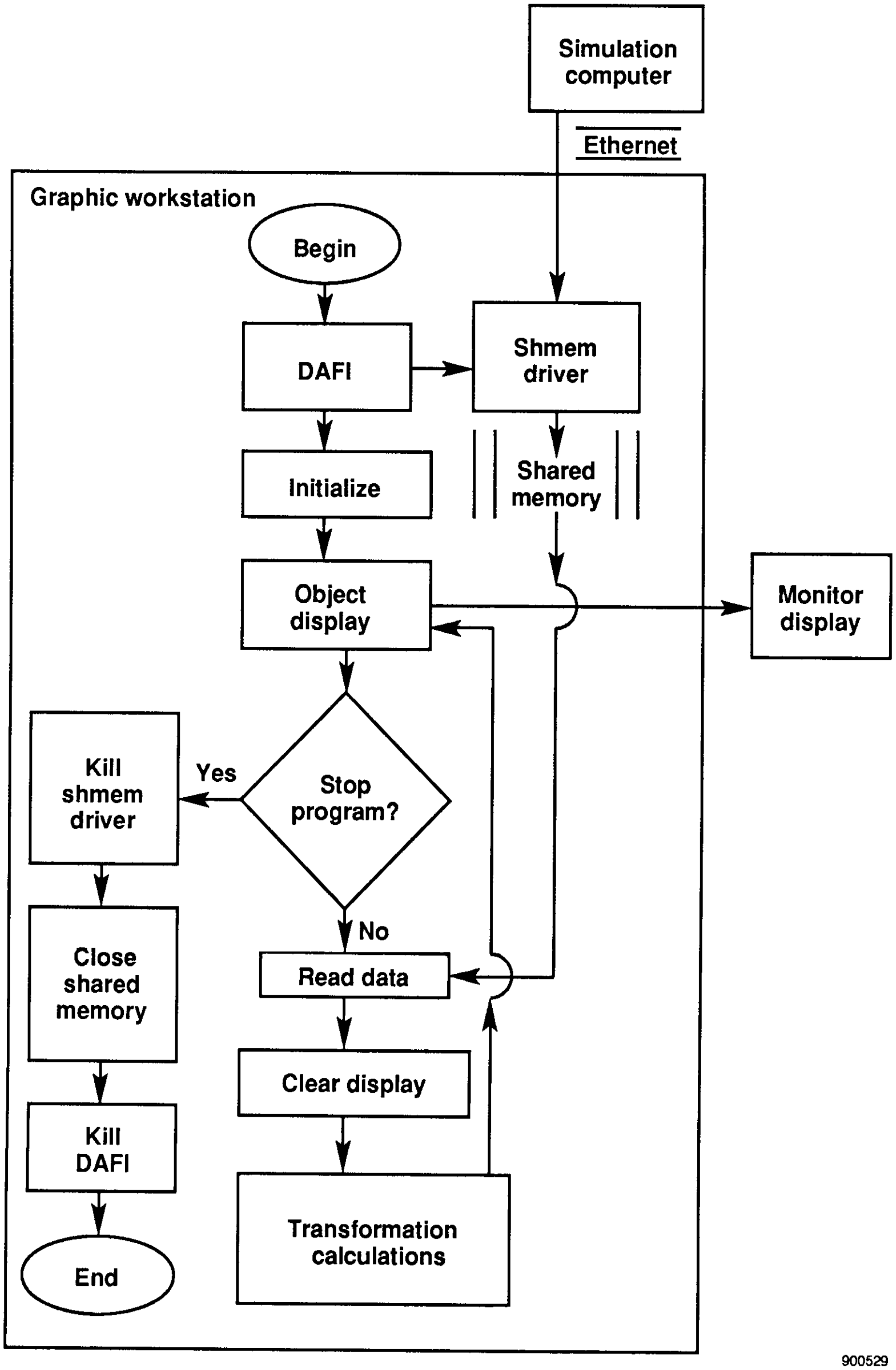


Figure 8. Program processing path.

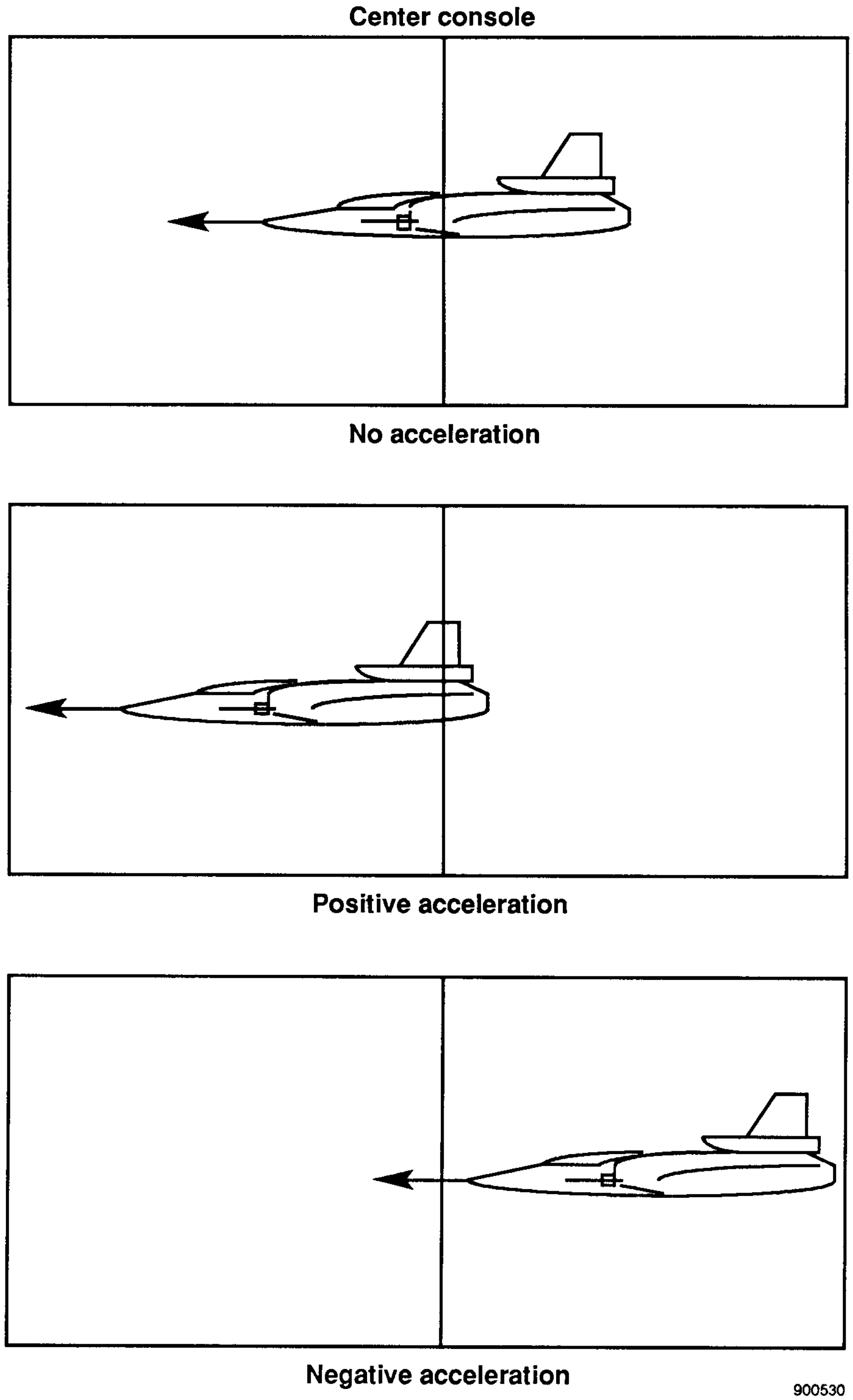


Figure 9. Position correlation with respect to acceleration.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No. NASA TM-101730	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Real-Time Application of Advanced Three-Dimensional Graphic Techniques for Research Aircraft Simulation		5. Report Date December 1990
		6. Performing Organization Code
7. Author(s) Steven B. Davis		8. Performing Organization Report No. H-1642
		10. Work Unit No. RTOP 505-68-27
9. Performing Organization Name and Address NASA Ames Research Center Dryden Flight Research Facility P.O. Box 273, Edwards, California 93523-0273		11. Contract or Grant No.
		13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-3191		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Visual aids are valuable assets to engineers for design, demonstration, and evaluation. This report discusses a variety of advanced three-dimensional graphic techniques used to enhance the displays of test aircraft dynamics. The new software's capabilities are examined and possible future uses considered.

17. Key Words (Suggested by Author(s)) Graphic display software, Dynamic modeling X-29 aircraft modeling, Real-time modeling, Graphical flight test analysis	18. Distribution Statement Unclassified-Unlimited Subject category 61

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 21	22. Price A02